

```

#R CMD INSTALL \\Aa.ad.epa.gov\ord\RTP\Users\E-J\jwambaug\NETMyD~1\Resear~1\vLiverPBPK

#setwd("\\\\Aa/ord/RTP/USers/E-J/jwambaug/NET MyDocuments/Research Projects/vLiverPBPK")

#Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^4))
#Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^3),CAS.end=37)
#Make.Wetmore.Plot(Wetmore,ExpoCast,ylim=c(10^-12,10^3),CAS.start=38,CAS.end=74)
#Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))
#Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,random.suppress=0.1,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))
#Make.Wetmore.Plot(Wetmore,predictions.lower.col=NULL,cex.compound=0.15,ylab="AC50(uM)",dose.lower.col="AC50",dose.order.col="Lower.Oral.Equivalent.mg.kg.",predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-5,10^4))
#Make.Wetmore.Plot(Wetmore,predictions.lower.col=NULL,cex.compound=0.15,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-5,10^4))
#Make.Wetmore.Plot(Wetmore,Wetmore.Exposures,predictions.lower.col=NULL,predictions.mean.col="Exposure.Estimates..mg.kg.day.",predictions.CAS.col="CASRN",predictions.nearfield.col=NULL,ylim=c(10^-12,10^4))

```

```
library(vLiverPBPK)
```

```

makeWetmoreCssPlot <- function(Wetmore.human.values,pred.human.values,Wetmore.rat.values,pred.rat.values,chemlines=F,title="Recreation of SimCYP Results",xlim=c(10^-3,10^3),ylim=c(10^-3,10^3),log="xy",xlab="Css Predicted",ylab="Css Wetmore et al. (2012,2013)") {
  plot(10^-6,10^-6,xlim=xlim,ylim=ylim,log=log,xlab=xlab,ylab=ylab)
  for (this.CAS in names(Wetmore.human.values)) {
    matplot(pred.human.values[[this.CAS]][1],Wetmore.human.values[[this.CAS]][1],pch=0,add=T)
    matplot(pred.human.values[[this.CAS]][2],Wetmore.human.values[[this.CAS]][2],pch=1,add=T)
    matplot(pred.human.values[[this.CAS]][3],Wetmore.human.values[[this.CAS]][3],pch=2,add=T)
    if (chemlines)
      matplot(c(pred.human.values[[this.CAS]][1],pred.human.values[[this.CAS]][2],pred.human.values[[this.CAS]][3]),c(Wetmore.human.values[[this.CAS]][1],Wetmore.human.values[[this.CAS]][2],Wetmore.human.values[[this.CAS]][3]),lty=3,add=T,type="l")
    }
    for (this.CAS in names(Wetmore.rat.values)) {
      matplot(pred.rat.values[[this.CAS]][1],Wetmore.rat.values[[this.CAS]][1],pch=16,add=T)
    }
    matplot(c(10^-4,10^4),c(10^-4,10^4),lty=2,add=T,type="l")
    matplot(90,10^-1,pch=0,add=T)
    matplot(90,10^-1.25,pch=1,add=T)
    matplot(90,10^-1.5,pch=2,add=T)
    matplot(90,10^-1.75,pch=19,add=T)
    text(100,10^-1,"Lower 95th",pos=4)
    text(100,10^-1.25,"Median",pos=4)
    text(100,10^-1.5,"Upper 5th",pos=4)
    text(100,10^-1.75,"Rat",pos=4)
  }
}

```

```

title(title)
}

makeCssInfsRDPlot <- function(inf.values, RD.values, set1.members, set1.label="Slow
Metabolism", set2.label="Rapid Metabolism", title="Comparison of Infusion and Repeated Dosing Steady
States", xlim=c(10^-3,10^3), ylim=c(10^-3,10^3), log="xy", xlab="Predicted Infusion Css", ylab="Predicted
Repeated Dosing Css")
{
  plot(10^-6,10^-6,xlim=xlim,ylim=ylim,log=log,xlab=xlab,ylab=ylab)
  for (this.CAS in names(inf.values[set1.members]))
  {
    matplot(inf.values[[this.CAS]][1],RD.values[[this.CAS]][1],pch=0,add=T)
  }
  for (this.CAS in names(inf.values[!set1.members]))
  {
    matplot(inf.values[[this.CAS]][1],RD.values[[this.CAS]][1],pch=15,add=T)
  }
  matplot(c(10^-4,10^4),c(10^-4,10^4),lty=2,add=T,type="l")
  matplot(5,10^-1,pch=0,add=T)
  matplot(5,10^-1.25,pch=15,add=T)
  text(6,10^-1,set1.label,pos=4)
  text(6,10^-1.25,set2.label,pos=4)
  title(title)
}

```

# Fig 1 SimCYP vs. our predictions:

```

vary.params <- list(BW=0.2)
vary.params[["liver.volume.per.kgBW"]]<-0.2
vary.params[["QGFRc"]]<-0.2
vary.params[["Qhc"]]<-0.2
vary.params[["million.cells.per.gliver"]]<-0.2
vary.params[["CLint"]]<-0.2

Wetmore.human.values <- NULL
vLiver.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_withlodplasma_CAS())
  {
    Wetmore.human.values[[this.CAS]] <-
      c(Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.05),Wetmore_Css(chem.CAS=this.CAS,which.qu
      antile=0.5),Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.95))
    these.params <- parameterize_SteadyState(this.CAS)
    if (these.params[["fub"]] == 0.0) these.params[["fub"]] <- 0.005
    vLiver.human.values[[this.CAS]] <-
      UVPKM("calc_Css",these.params,CV.params=vary.params,which.quantile=c(0.05,0.5,0.95))
  }

Wetmore.rat.values <- NULL
vLiver.rat.values <- NULL
for (this.CAS in get_Wetmore_CAS("Rat"))
  if (this.CAS %in% get_Css_withlodplasma_CAS("Rat"))
  {
    Wetmore.rat.values[[this.CAS]] <-
      Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5,species="Rat")
    these.params <- parameterize_SteadyState(this.CAS,species="Rat")
  }

```

```

if (these.params[["fub"]] < 0.01) these.params[["fub"]] <- 0.005
vLiver.rat.values[[this.CAS]] <- calc_Css(these.params)
}

```

```

makeWetmoreCssPlot(Wetmore.human.values,vLiver.human.values,Wetmore.rat.values,vLiver.rat.values
,chemlines=T)

```

# Fig 2 SimCYP vs. our predictions with censored Fub:

```
censored.params<-list(fub=list(sd=0.2, lod=0.01))
```

```

Wetmore.fub.human.values <- NULL
vLiver.fub.human.values <- NULL
for (this.CAS in get_Wetmore_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    Wetmore.fub.human.values[[this.CAS]] <-
      c(Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.05),Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5),Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.95))
    these.params <- parameterize_SteadyState(this.CAS)
    vLiver.fub.human.values[[this.CAS]] <-
      UVPKM("calc_Css",these.params,CV.params=vary.params,censored.params=censored.params,which.quantile=c(0.05,0.5,0.95))
  }

```

```

Wetmore.fub.rat.values <- NULL
vLiver.fub.rat.values <- NULL
for (this.CAS in get_Wetmore_CAS("Rat"))
  if (this.CAS %in% get_Css_CAS("Rat"))
  {
    Wetmore.fub.rat.values[[this.CAS]] <-
      Wetmore_Css(chem.CAS=this.CAS,which.quantile=0.5,species="Rat")
    these.params <- parameterize_SteadyState(this.CAS,species="Rat")
    vLiver.fub.rat.values[[this.CAS]] <-
      UVPKM("calc_Css",these.params,censored.params=censored.params,which.quantile=0.5)
  }

```

```

makeWetmoreCssPlot(Wetmore.fub.human.values,vLiver.fub.human.values,Wetmore.fub.rat.values,vLiver.fub.rat.values,title="Fub Left-Censored at Limit of Detection",chemlines=T)

```

#Fig 3 Infusion Css vs. Daily dosing PBPK Css

```

infusion.human.values <- NULL
dailydosing.human.values <- NULL
CLint.values <- NULL
for (this.CAS in get_PBPK_CAS())
  if (this.CAS %in% get_Css_CAS())
  {
    these.params <- parameterize_SteadyState(this.CAS)
    infusion.human.values[[this.CAS]] <- calc_Css(these.params)
  }

```

```
these.params <- parameterize_vLiverPBPK(this.CAS)
dailydosing.human.values[[this.CAS]] <- calc_Css(these.params,doses.per.day=3)
CLint.values[[this.CAS]] <-
chem.physical_and_invitro.data[chem.physical_and_invitro.data$CAS==this.CAS,"Human.Clint"]
}

set1 <- as.numeric(unlist(CLint.values))>median(as.numeric(unlist(CLint.values)))
makeCssInfvsRDPlot(infusion.human.values,dailydosing.human.values,set1)
```